REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
5/8/95

3. REPORT TYPE AND DATES COVERED
Semi-Annual 10/31/94 - 4/4/95

4. TITLE AND SUBTITLE
Numerical and Symbolic Algorithms for Application Specific Signal Processing

5. FUNDING NUMBERS
N00014-93-1-0686
rassp01-01

6. AUTHOR(S)
Prof. Alan V. Oppenheim

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Research Laboratory of Electronics
Massachusetts Institute of Technology
77 Massachusetts Avenue
Cambridge, MA 02139

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research
Ballston Tower One
800 North Quincy Street
Arlington, VA 22217-5660

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The view, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Work by Prof. Oppenheim and his collaborators is summarized here

14. SUBJECT TERMS

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Semi-Annual Report

Numerical and Symbolic Algorithms for Application Specific Signal Processing

October 31, 1994 – April 4, 1995

Research Organization: Digital Signal Processing Group
Research Laboratory of Electronics
Massachusetts Institute of Technology

Principal Investigator: Alan V. Oppenheim
Distinguished Professor of Electrical Engineering

Grant Number: N00014-93-1-0686
OSP Number: 60314

Program Manager: Mr. Clifford Lau

1 Introduction

During the period of this grant, our detailed technical accomplishments are reported through journal articles and technical reports. Each of our semi-annual reports will highlight certain technical areas and provide a summary listing of our technical articles related to the project.

2 Algorithm-Based Fault Tolerance

Our work under RASSP in the area of Algorithm-Based Fault Tolerance has concentrated on generalizations of arithmetic coding schemes. The starting point for this research is the 1993 doctoral thesis of Beckmann at MIT [1]. We have been able to obtain some interesting and important extensions, documented in the Masters thesis of Hadjicostis [2], which is available on request.

Beckman's most detailed results were for the case of computations occurring in an algebraic group. He established that coding for fault tolerance involved mapping from an original group to a larger group via a group homomorphism. The redundancy in the larger group is responsible for the fault tolerance. Beckmann also examined algebraic structures – such as rings – that have an embedded group operation (actually [2] clarifies some of the arguments of [1] for the case of rings). His framework is able to embrace most known schemes for arithmetic coding, all the way from direct redundancy to, for example, various parity-based schemes that involve residue arithmetic.

Our first generalization under RASSP was to the case of computations occuring in semigroups. In this setting, one loses some of the possibilities that are available with groups, but much of the group framework survives. In particular, fault tolerance is obtained by using a semigroup homomorhism to map to a larger, redundant semigroup. We have obtained a full characterization, in the semigroup case, of all possible "separate" codes, i.e. codes that involve a separate parity channel. For the ring of integers, the basic result goes back to Peterson in 1958, who showed that the only possibility was a residue computation. Beckmann's extension showed that all possible separate codes for group computations could be determined by listing all possible subgroups of the original group. In our semigroup case, the characterization involves the determination of all possible congruence relations on the original semigroup.

The semigroup examples treated in [2] include the case of nonnegative integers under addition, $(N0, +)$, and the positive integers under multiplication, $(N, *)$. In the former case, all possible parity check codes have been explicitly identified, while in the latter case some possibiliites have been described in detail.

The more interesting algebraic structures in signal processing involve two operations rather than one (which is why rings are more prevalent than groups). The natural setting in which to embed our semigroup results seems to be that of semirings. A semiring differs from a ring in that the "addition" operation is a semigroup operation rather than a group operation. Examples of semirings include: $(N0, +, *)$, the set of nonnegative integers under integer addition and multiplication; $(Z, \max, +)$, the set of integers (actually augmented with negative infinity), with max being the "additive" operation and $+$ (i.e., normal integer addition) being the "multiplicative" operation; $(Z, \max, \min)$; and so on. Several of the structures used in nonlinear signal processing schemes of current interest appear to involve semirings.

We have obtained natural refinements of our semigroup results for the case of semirings. As might be expected, semiring homomorphisms from the original semiring to a larger one play an important role. All possible separate (i.e. parity-based) codes are identified by listing all the semiring congruence classes in the original semiring. One can immediately use these results to show, for example, that the only possible arithmetic codes for $(Z, \max, +)$ must be non-separate.

Our results on semirings have only scratched the surface of what seems possible and useful here, so we expect to continue in this direction in the near future. While most of our results have been developed with the assumption of cummutativity, this restriction is not really needed throughout (and [2] does include results for the noncommutative ring of n-by-n matrices, for instance); we shall treat the noncommutative case more fully in future work. We also intend to examine the protection of sequences or strings of computations, rather than individual computations. Another significant challenge for work in the coming year is to connect the algebraic framework with the specifics of actual hardware realizations and their potential failures.

3 I/Q Conversion and Low-Power Signal Processing

In this section we describe a scheme for I/Q conversion.

Many techniques for I/Q conversion have been proposed. Most of them focus on ways to reduce the computation in time domain implementations of the filtering required on each channel. In a paper which will be presented in July at SPIE 95 in San Diego, we describe a method for saving computation in a frequency-domain implementation of the filtering on each channel. The scheme makes it possible to filter both channels at once using just one FFT and one IFFT. In this scheme the FFT routines must accept complex data. The ideas for this work arouse in connection with a graduate student's summer work on the design of a synthetic aperture radar system for the Lockheed Sanders RASSP program.

The rapidly growing demand for portable electronic systems has led to a desire to design these systems so as to consume as little power as possible. This is bringing a paradigm shift in how engineers view performance. Whereas MOPS had been the chief figure of merit, now MOPS/Watt is becoming increasingly important. Power consumption in CMOS circuits is directly proportional to switching activity. If the circuit doesn't switch, then power is not consumed. This opens the door to consider many techniques at the algorithm and architecture level which seek to minimize the expected switching activity needed to perform a given operation. This adds an entirely new dimension beyond simply counting operations, since transition activity on buses and in adders and multipliers will be affected by signal statistics. We are currently considering how to find computational structures which will consume the least expected power given some set of assumptions. There clearly are many ways to approach this problem depending on the signal processing task at hand, the a priori knowledge available, etc., and part of the challenge is to investigate the wide variety of ways of formulating the problem.

Since the I/Q conversion work does not seem to be in need of further development, our future work on the topics mentioned here will focus on more investigation of algorithms for low-power signal processing. There are many unexplored avenues in this area, and we hope to find the fruitful destinations in the months ahead.

4 Low-Power Signal Processing

An important thrust in our research under RASSP is aimed at a design framework for low-power signal processing. It is clear that a major factor in the weight and size of portable electronic systems is the amount of batteries

which is directly impacted by the power dissipated by the electronic circuits. In addition, the cost of providing power (and associated cooling) has resulted in significant interest in power reduction even for non-portable applications that have access to a power source. In spite of these concerns, until recently, there has not been a major focus on a digital circuit design methodology that directly addresses power reduction; instead, the focus has been on ever faster clock rates and logic speeds. The strict limitation on power dissipation which portability imposes must be met by the designer while still meeting even higher computational requirements. This is resulting in a need for "Cold Chip" design strategy. To meet this need, a comprehensive approach is required at all levels of the system design, ranging from algorithms and architectures to the logic styles and the underlying technology. The goal of our research suited for the low-power implementation of Digital Signal Processing applications. We are also focusing on the CAD tools required to explore the design space at high levels of abstraction (algorithm and architecture) and minimize power.

Power in digital CMOS circuits is primarily consumed in charging and discharging parasitic capacitors and is given by:

$$P = \sum N_i C_i V_{dd}^2 \text{fsample} \tag{1}$$

where C_i is the capacitance switched to perform operation i (representing multiplications, additions, bus accesses, etc), N_i is the number of times operation i is performed per sample period, V_{dd} is the power supply voltage, and fsample is the sample frequency. In order to minimize the power consumption, the various components of power must be minimized. In this work we will assume that the application to be implemented in low power is known, and trade-offs can be made as long as the functionality required of this application is met within a given time constraint.

Maintaining a given level of computation or throughput is a common concept in signal processing and other dedicated applications, in which there is no advantage in performing the computation faster than some given rate, since the processor will simply have to wait until further processing is required. That is, fsample is fixed for a particular application (e.g. a video coder has to compress a video frame in 30ms). This is in contrast to general purpose computing, where the goal is often to provide the fastest prossible computation without bound. One of the most important ramifications of only maintaining throughput is that it enables an architecture driven voltage scaling strategy, in which aggressive voltage reduction is used to reduce

power, and the resulting reduction in logic speed (since gate delays in CMOS circuits increase with a reduction in supply voltage) is compensated through parallel architectures to maintain throughput [3]. However, this technique is also applicable to the general purpose environment, if the figure of merit is the amount of processing per unit of power dissipation (e.g. MIPS/watt) since in this case the efficiency in implementing the computation is considered and voltage scaling decreases the energy expended per evaluation.

Since CMOS circuits do not dissipate power if they are not switching, a major focus of low power design is to reduce the switching activity to the minimal level required to perform the computation. This can range from simply powering down the complete circuit or portions of it, to more sophisticated schemes in which the clocks are gated or optimized circuit architectures are used which minimize the number of transitions. An important attribute which can be used in circuit and architectural optimization is the correlation which can exist between values of a temporal sequence of data, since switching should decrease if the data is slowly changing (highly correlated).

At the algorithm level, it is possible to minimize the number of switching events by intelligent choice of algorithm. For example, choosing tree search vector quantization over full-search vector quantization can reduce the computational requirements significantly. For a video compression module, assuming a block size of 4x4 and a 256 level codebook, the number of operations (and hence the switched capacitance) can be reduced by a factor of 16. Some of the other ways to reduce the computational complexity at the algorithmic level include substituting multiplications with constants to shift-add operations, scaling filter coefficients for minimal number of shift-adds, optimizing bit-width, choice of data representation, etc.

The choice of data representation can have a significant impact on the power. In most signal processing applications, two's complement is typically chosen to represent numbers since arithmetic operations (addition and subtraction) are easy to perform. However, one of the problems with two's complement representation is sign-extension, which causes the MSB sign-bits to switch when a signal transitions from positive to negative or vice-versa (for example, going from -1 to 0 will result in all of the bits toggling). Therefore using a two's complement representation can result in significant switching activity when the signals being processed switch frequently around zero and when they do not utilize the entire bit-width (i.e., the dynamic range is much smaller than the maximum possible value determined from the bit-width) since a lot of the MSB bits will perform sign-extension. Even if a

signal utilizes the entire bit-width, arithmetic operations such as scaling can reduce the signal dynamic range. One approach to minimizing the switching in the MSBs is to use a sign-magnitude representation, in which only one bit is allocated for the sign and the rest for the magnitude. In this case, if the dynamic range of a signal does not span the entire bitwidth, only one bit will toggle when the signal switches sign, as opposed to the two's complement representation where due to sign extension several of the bits will switch. We are also investigating other data representations reducing switching activity.

We have recently been investigating approaches for power reduction in digital CMOS filter design using approximate processing techniques [7]. The basic idea is to adaptively reduce the number of operations switched per sample based on signal statistics. We have focused so far on the low-power design of approximate processing filters [6]. Finite impulse response (FIR) filters are often used in applications where the goal is to extract from a signal certain frequency components while rejecting others. For example, in order to receive a single channel of a frequency division multiplexed (FDM) signal, the frequency band containing the channel of interest is passed by a frequency-selective filter while other bands are attenuated. Suppose the signal to be filtered is the sum of a desired signal, $s[n]$, and an interference or noise component, $w[n]$:

$$x[n] = s[n] + w[n] \tag{2}$$

many contexts arise in which the signals $s[n]$ and $w[n]$ occupy largely disjoint frequency bands. If it were possible to cost-effectively measure the strength of the interference, $w[n]$, from observation of $x[n]$, we could determine how much stopband attenuation the filter should have at any particular time. Therefore, when the energy in $w[n]$ increases, the stopband attenuation of the filter is also increased. This can be accomplished by using a longer FIR filter. Conversely, the filter may be made shorter when the energy in $w[n]$ decreases. Powering down of the higher order taps has the effect of reducing the switched capacitance at the cost of decreasing the attenuation in the stopband. Assuming that the delay line is implemented using SRAM, even the data shifting operation of the higher order taps can be eliminated through appropriate addressing schemes.

The key to being able to adaptively adjust the filter length is to somehow obtain a measure of the strength of the interference $w[n]$ in a cost-effective manner since this is overhead circuitry consuming power. For example, if

signal being filtered is approximately equal to the interference signal, we can use $x[n]$ to measure the strength of $w[n]$. In particular, the measure we have used for this purpose is given by:

$$M_x[n] = \frac{1}{L} \sum_{m=n-L+1}^{L} (|x[m]|) \tag{3}$$

We selected this measure because of its direct proportionality to the local energy in $x[n]$ and because its computation does not require any multiplication operations. Furthermore, only two additions are required to obtain $Mx[n+1]$ from $Mx[n]$. The choice of the parameter L involves a trade-off between suppression of sensitivity to local fluctuations and preservation of time-varying nature of the signal energy. When the value of L is less than the maximum filter length, then there is no extra storage required to compute $Mx[n]$.

Our initial results using the above appoximate processing techniques indicate that the power consumption can be potentially reduced over conventional solutions by an order of magnitude for wireless applications. We are currently looking at other approaches to reduce power in filters using approximate processing techniques which involve more sophisticated methods for estimating the amount of filtering required. These concepts are also being applied to other contexts such as speech and video coding. Efficient architectures suited for approximate processing are also being investigated.

Another component of our low-power research effort is the development of CAD tools that can be used to automatically search the design space and find computational structures with the lowest power consumption. As a part of the InfoPad project [4], a transformation based approach to minimize power has been implemented in HYPER, a high-level synthesis system [5]. HYPER (which is integrated into Ptolemy) takes a high level specification of an algorithm and optimizes the design using computational transformations. The synthesis environment consists of high-level estimation of power consumption, a library of transformation primitives, and heuristic/probabilistic optimization search mechanisms for fast and efficient scanning of the design space.

High-level power estimation involves estimating the power consumption of a design from a high-level description (like C, silage or VHDL). Conventional approaches to power estimation fall under three categories: gate-level probabilistic estimation, switch-level estimation, and circuit-level estimation. The primary trade-off between these approaches is the computational

complexity vs. accuracy. These approaches estimate power consumption from a low-level of abstraction. To use these approaches in a high-level synthesis framework, the high-level representation of the algorithm has to be mapped to a low-level description (gate or transistor level), which is very time consuming. Also, the estimation time for each new topology is too long to meaningfully explore many architectures. Hence, power must be estimated efficiently from a high level of abstraction. HYPER estimates power from an algorithmic level so the design space can be quickly explored. Power estimation involves estimating power consumed in the execution units, memory, interconnect and control. A combination of analytical models and statistical models is used. For example, building a model for interconnect involves taking into account the effects of various synthesis and layout tools. An extensive experimental study, followed by in-depth statistical analysis and verification is the only viable solution which will satisfy the contradictory requirements of modeling a complex system, with high accuracy in a computationally efficient manner. The model for interconnect capacitance was built using fifty examples which were mapped from their high-level descriptions to layout.

The tool uses various transformations (retiming, loop-unrolling, algebraic transformations, pipelining, etc.) to minimize the power supply voltage and the switched capacitance. We are planning to extend the tool to include optimization involving approximate procesing, filtering coefficient selection, data representation, bit-width, etc.

5 References

[1] P.E. Beckmann, "Fault-Tolerant Computation using Algebraic Homomorphisms," RLE Tech. Rep. 580, Research Laboratory of Electronics, MIT, June 1993.

[2] C. Hadjicostis, "Fault-Tolerant Computation in Semigroups and Semirings," M.Eng. thesis, EECS Department, MIT, February 1995.

[3] A. P. Chandrakasan, S. Sheng, R. W. Brodersen, "Low-power digital CMOS design," *IEEE Journal of Solid State Circuits*, pp. 473-484, April 1992.

[4] A. P. Chandrakasan, A. Burstein, R. W. Brodersen, "A Low-power Chipset for a Portable Multimedia I/O Terminal," *IEEE Journal of Solid-State Circuits*, pp. 1415-1428, December 1994.

[5] A. P. Chandrakasan, R. Mehra, M. Potkonjak, J. Rabaey, R. W. Brodersen, "Optimizing Power Using Transformations," accepted for publication in the *IEEE Transaction on CAD*, January 1995.

[6] J. Ludwig, S. H. Nawab, A. P. Chandrakasan, "Low Power Filtering Using Approximate Processing for DSP Applications", submitted for the 1995 Custom Integrated Circuits Conference.

[7] A. V. Oppenheim, S. H. Nawab, G. Verghese and G. Wornell, "Algorithms for Signal Processing," Proceedings of the First Annual RASSP Conference, Washington, D.C., August 15-19, 1994.

6 Publications of Work Supported

[1] Arthur B. Baggeroer and Henrik Schmidt, "Cramer-Rao Bounds for Matched Field Tomography and Ocean Acoustic Tomography," IEEE International Conference on Acoustic, Speech and Signal Processing, ICASSP-95, Detroit, May 1995.

[2] Kevin M. Cuomo, "Systematic Synthesis Procedures for High-Dimensional Chaotic Systems," IEEE International Symposium on Circuits and Systems, Seattle, April 1995.

[3] C. Hadjicostis, "Fault-Tolerant Computation in Semigroups and Semirings," M.Eng. thesis, EECS Department, MIT, February 1995.

[4] Steven H. Isabelle, "A Signal Processing Framework for the Analysis and Application of Chaotic Systems," Ph.D. thesis, EECS Department, MIT, February 1995.

[5] Steven H. Isabelle and Gregory W. Wornell, "Statistical Properties of One-Dimensional Chaotic Signals," IEEE International Conference on Acoustic, Speech and Signal Processing, ICASSP-95, Detroit, May 1995.

[6] J. Ludwig, S. H. Nawab, A. P. Chandrakasan, "Low Power Filtering Using Approximate Processing for DSP Applications", submitted for the 1995 Custom Integrated Circuits Conference.

[7] S. Hamid Nawab and Joseph M. Winograd, "Approximate Signal Processing Using Incremental Refinement and Deadline-Based Algorithms,"

IEEE International Conference on Acoustic, Speech and Signal Processing, ICASSP-95, Detroit, May 1995.

[8] Andrew C. Singer, "Signaling Techniques Using Solitons," IEEE International Conference on Acoustic, Speech and Signal Processing, ICASSP-95, Detroit, May 1995.

[9] Joseph M. Winograd and S. Hamid Nawab, "Mixed-radix approach to incremental DFT refinement," SPIE '95 Conference, San Diego.

[10] Joseph M. Winograd and S. Hamid Nawab, "A C++ Software Environment for the Development of Embedded Signal Processing," IEEE International Conference on Acoustic, Speech and Signal Processing, ICASSP-95, Detroit, May 1995.

[11] Joseph M. Winograd and S. Hamid Nawab, "Incremental Refinement of DFT and STFT Approximations," *Signal Processing Letters*, February 1995, Vol. 2, No. 2, pp. 25-27.

ATTACHMENT NUMBER 2

REPORTS AND REPORT DISTRIBUTION

REPORT TYPES

(a) Performance (Technical) Report(s) (Include letter report(s))
Frequency: Semiannual

(b) Final Technical Report, issued at completion of Grant.

(c) Final Financial Status Report (SF 269)

REPORT DISTRIBUTION

ADDRESSEES	REPORT TYPES	NUMBER OF COPIES
SCIENTIFIC OFFICER CODE: 1114SE Clifford G. Lau OFFICE OF NAVAL RESEARCH BALLSTON TOWER ONE 800 NORTH QUINCY STREET ARLINGTON, VIRGINIA 22217-5660	(a) & (b)	3
ADMINISTRATIVE GRANTS OFFICER OFFICE OF NAVAL RESEARCH RESIDENT REPRESENTATIVE ONR MIT 495 SUMMER STREET ROOM 103 BOSTON MA 02210-2109	(a) & (b) & (c)	1
DIRECTOR, NAVAL RESEARCH LABORATORY ATTN: Code 2627 WASHINGTON, DC 20375	(a) & (b)	1
DEFENSE TECHNICAL INFORMATION CENTER BUILDING 5, CAMERON STATION ALEXANDRIA, VIRGINIA 22304-6145	(a) & (b)	2

If the Scientific Officer directs, the Grantee shall make additional distribution of technical reports in accordance with a supplemental distribution list provided by the Scientific Officer. The supplemental distribution list shall not exceed 250 addresses.